Rule CIC190: The number of DL/I Threads may be too low

Finding: CPExpert believes that the number of DL/I Threads may be too low.

Impact: This finding should normally have a LOW IMPACT or MEDIUM IMPACT on the performance of the CICS region.

Logic flow: This is a basic finding, based upon an analysis of the CICS statistics.

Discussion: The DLTHRED operand in the System Initialization Table (SIT) specifies the number of concurrent DL/I threads that can be allocated for IMS/VS data bases. The number of concurrent DL/I threads limits the number of tasks concurrently scheduled for use of IMS/VS resources.

The DLTHRED operand is used at IMS/VS initialization to create control blocks. One set of control blocks is created for each DL/I thread. For each DL/I thread, control blocks are allocated for the CICS Interface Scheduling Block (ISB), the IMS/VS Parameter Block (PXPARMS), and the Partition Specification Table (PST). Storage for these control blocks is acquired at CICS initialization. The control blocks are retained while CICS is operational.

The amount of storage required for each DL/I thread depends upon the release of IMS. For each DL/I thread, IMS/VS 2.2 requires 9K bytes of storage and IMS/ESA requires 12K bytes of storage. A significant amount of real and virtual storage can be required if a large number of DL/I threads are specified. However, if the number of DL/I threads is too low, tasks can wait for threads. If tasks wait for DL/I threads, they tie up storage while they are waiting, and response suffers. Therefore, there is a tradeoff between the amount of storage allocated for DL/I threads and the delays caused by tasks waiting for threads.

CPExpert produces Rule CIC190 if storage was not a constraint and the number of times tasks waited for DL/I Threads was greater than the DLIWAIT guidance value. The default value of the DLIWAIT guidance parameter is zero.

Suggestion: CPExpert suggests that you consider increasing the number of DL/I Threads, as specified by the DLTHRED operand in the SIT. Tasks should not wait for DL/I threads unless storage is a constraint.

In any event, the DLTHRED value should be less than the AMXT value in the SIT[1]. Serious system degradation can occur if the DLTHRED value is higher than the AMXT value. (Refer to Rule CIC191 for an explanation of this situation.)

If storage **is** a constraint to the CICS region, you may wish to leave the DLTHRED value at its current setting. Depending upon the seriousness of the storage constraint, you may even wish to decrease the DLTHRED value so that some tasks wait on DL/I threads.

The CICS Performance Guides indicate that it may be acceptable for 5-10% of the tasks to wait on DL/I threads. For most environments, CPExpert believes that the 5-10% value is much too high. However, you may have a unique situation; if you **do** have a unique situation, you should change the CPExpert DLIWAIT guidance variable to prevent the spurious firing of this rule.

NOTE: The significance of this finding depends upon whether the finding is based upon analyzing daily information or based upon analyzing historical information.

- If this finding is based upon an analysis of daily information, the finding may be applicable only to the performance of CICS for this day. Unless you feel that the analysis is generally applicable (or unless the workload processed on this day is particularly critical), please wait until CPExpert performs an analysis of historical information before taking action.

- If this finding is based upon an analysis of historical data covering a prolonged period, the finding is more definite than a tentative finding based upon analysis of only a single day's data.

Reference: *CICS/OS/VS Version 1.7 Performance* Guide: page 69 and pages 255-256.

CICS/MVS Version 2.1.2 Performance Guide: pages 180-181 and pages 398-399.

CICS/ESA Version 3.1.1 Performance Guide: pages 55-60 and pages 249-250.

CICS/ESA Version 3.2.1 Performance Guide: pages 170-171 and pages 275-280.

---

[1]This comment does not apply with CICS/ESA Version 4.1, as the AMXT parameter is not used with this level of CICS.

CICS/ESA Version 3.3.1 Performance Guide: page 181 and pages 295-299.

CICS/ESA Version 4.1.1 Performance Guide: Section 4.5.4 and Appendix A.1.5.

CICS/TS: not applicable.

CICS/TS for z/OS: not applicable. |

Rule CIC190.3